Introduction to TreeNet Modeling:
Comparing Accuracy of Earnings Predictions

Zachary Tausanovitch and Joyce Jacobsen, QAC Apprenticeship Summer 2011

Introduction to TreeNet

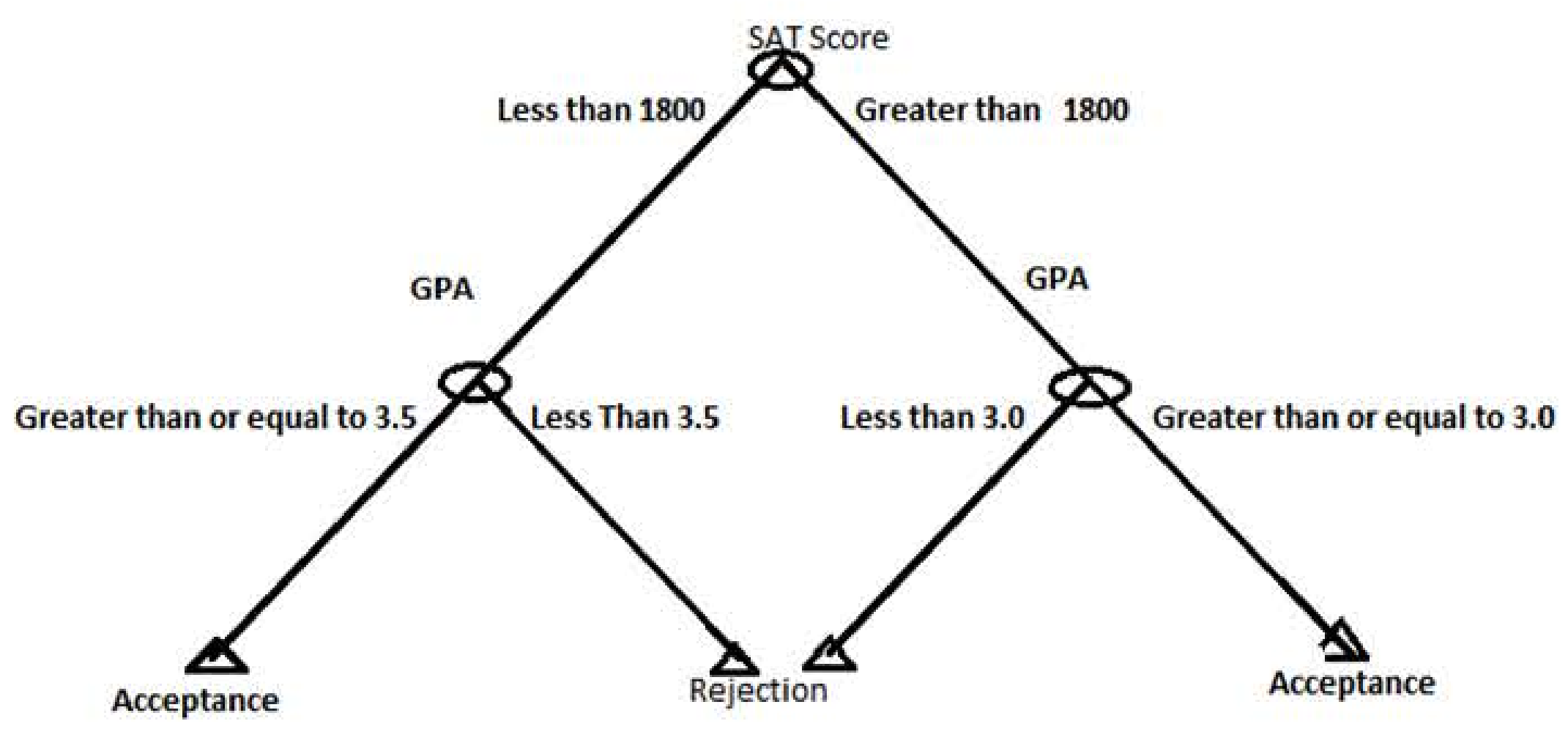
TreeNet is a program designed to execute a single algorithm. This algorithm is what some academics refer to as an ensemble model. This means that TreeNet doesn't estimate the model once, but many times using slightly adjusted models. The goal is to take the best qualities of each model, and combine them into a single ensemble model, similar to a neural net.

The estimation is based on models using decision trees. For a simplified example, consider a binary model that asks the question; what determines whether or not you are accepted to Wesleyan? In the first model consider only SAT score as an independent variable. While a regression would give you a coefficient, a decision tree would give you nodes such as: scores higher than 1800 mean acceptance; lower means rejection. This is a two terminal node tree, which branches at a single decision node. We could force this model to have more nodes, although it would likely have the same outcome, or possibly it could say below 1800 rejection, above 1800, if also below 2300 then acceptance, if above 2300 rejection. This tree has 3 terminal nodes, 2 decision nodes. If we have more than one explanatory variable the tree can only use one of them at any decision node, but may reuse them at separate nodes as many times as necessary to fit the model. So if we also add GPA, the model may look like: if SAT above 1800, and GPA > 3.0 or if GPA <3.0 and SAT >2100 then acceptance, if below 1800 and GPA>3.5 then acceptance, else, rejection. The question then becomes: How does TreeNet combine these models, and why is there more than one model?

TreeNet uses an optimizing technique called boosting. A simple example of boosting is when a model is built upon a random subset of data. For instance, 20% of the data is randomly selected for each model; in this way it is as if we are sampling from one population many times and building many estimates. Each model will be slightly different. Initially a decision tree is built on all the data to form a base model. Then TreeNet builds another decision tree on a random sampling of the data, and moves the estimates of the original tree closer to the new tree by adding more nodes. By repeating this process with possibly hundreds more trees our original few-node model now with possibly hundreds of nodes starts to look much more like a continuous curve.

A decision tree will eventually fit perfectly to a single set of data; we refer to this as over-fitting. Imagine a tree with a node for every data point; then the tree would report, for a given value a specific observation. This model would be useless for any other data set. To avoid this TreeNet builds the model only on some of the data and tests it on another portion that it does not use to build the trees. Using a loss function, TreeNet determines how far apart the estimates become and when they diverge too much TreeNet determines that the optimal number of trees has been reached. Also, by setting a minimum number of observations for each terminal node (implying that a split can only be made if the categories have enough people in each one), that one-to-one correspondence will become impossible.

Results of a TreeNet Estimation

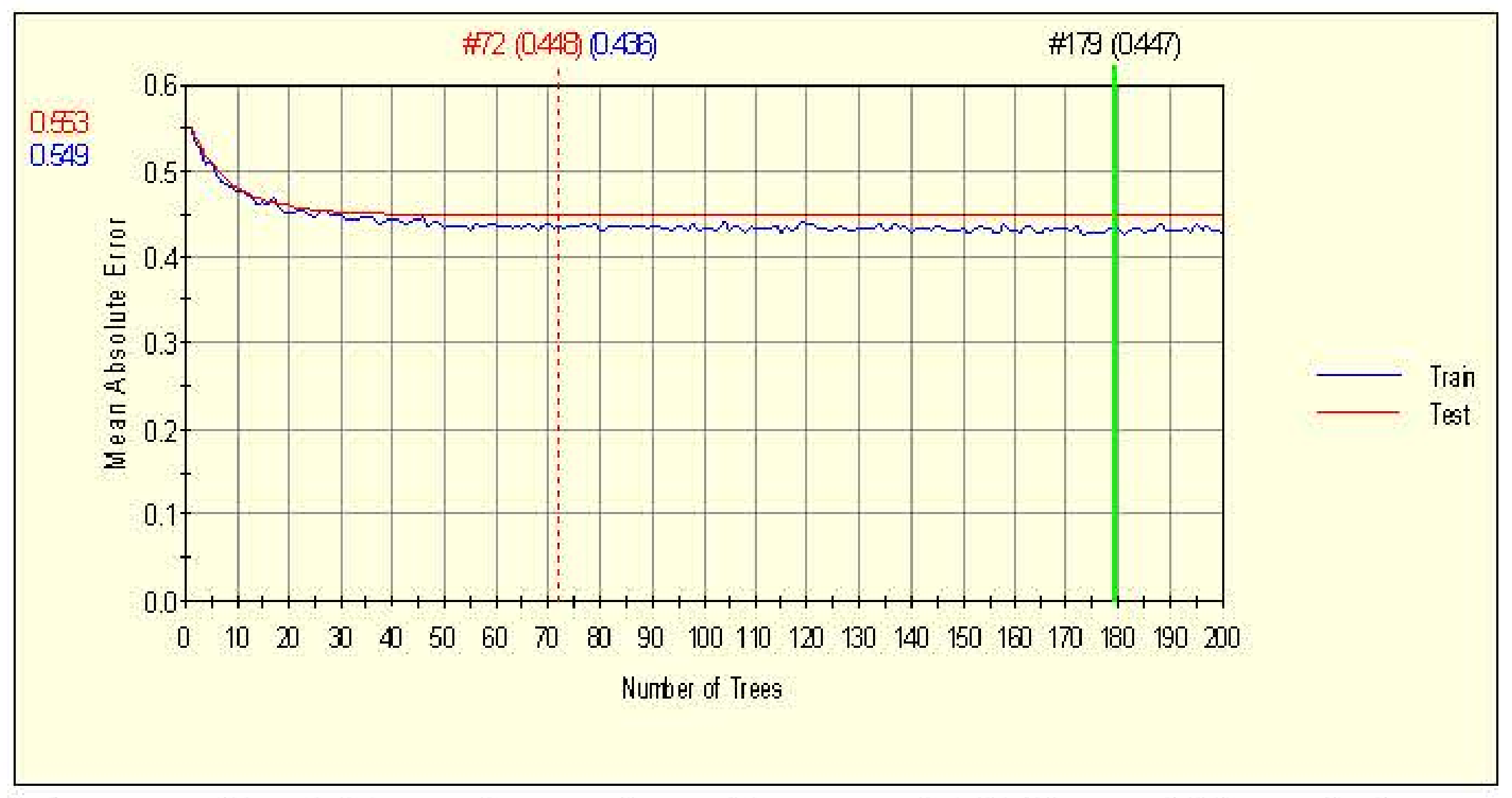


Results:

We are provided with several ways to evaluate our model: R-squared; learning and testing error; variable importance interaction plots and the gains chart. We are first presented with the R-squared and the learning and testing error.

Learning and Testing Error:

This graph shows the divergence of error between the training and the testing groups. The less error for both models the better, also the less they diverge the more externally valid the model will be. If the training error is fluctuating too much, this can also be an indicator of low external validity.



The green line indicates the number of trees at which the model has a balance between low error and low divergence test results; this is what is reported as the optimal number of trees.

Evaluating Variables:

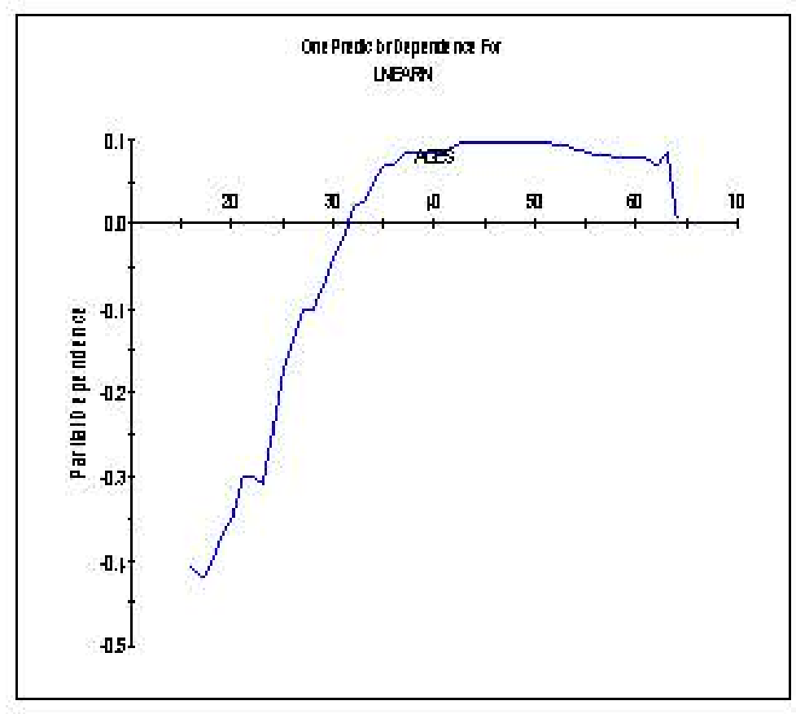
TreeNet cannot provide coefficients or t-scores, as it doesn't produce any. Instead we have Variable Importance charts, and Interaction plots. These are from a model that predicts earnings:

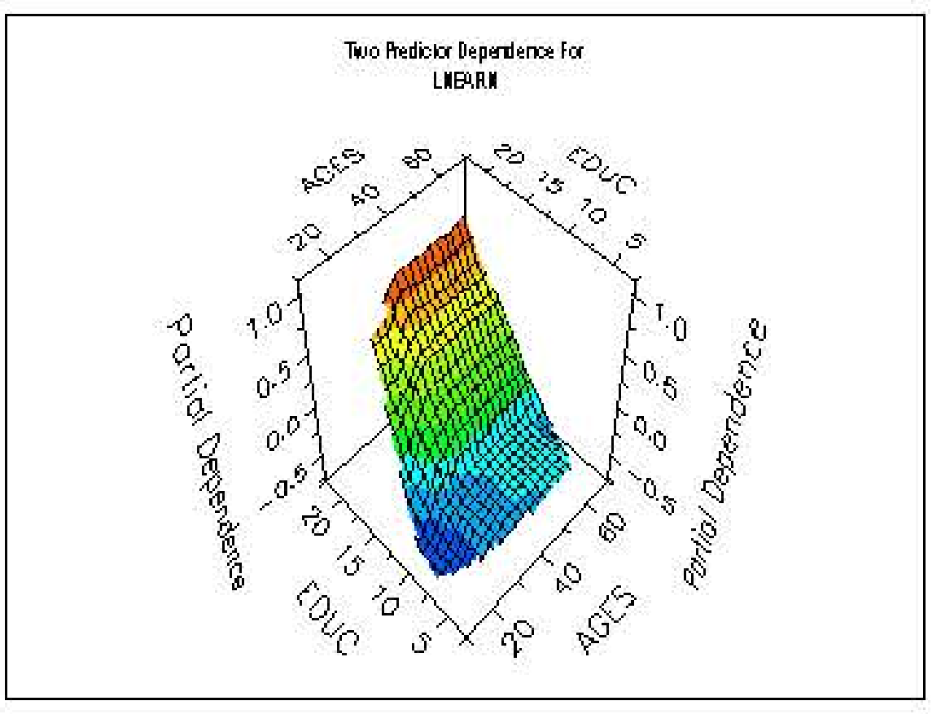
Variable Importance Plot

Variable	Score	
Education	100.00	||||||||||||||||||||||||||||||||||||||||
Age	63.83	|||||||||||||||||||||||||
Child support $ per year	56.82	||||||||||||||||||||||
Social security $ per year	41.29	||||||||||||||||
Gender	33.25	|||||||||||||
Marriage Status	29.76	||||||||||||
Country of Birth	18.08	|||||||
Race (white dummy)	16.29	||||||
Experience	16.11	||||||
Born in English country	8.82	|||
From the South	8.37	|||
Other income	4.63	|
Financial assistance income	2.91	
Age Squared	0.00	

The results of this table are normalized to the most important variable, which is the variable that has the largest effect on the tree results; in this case that variable is education. Notice that financial assistance is relatively unimportant. Dropping low importance variables may improve the model. TreeNet already considers non-linear relationships and thus age squared is already accounted for..

Interaction Plots: these graphs show at certain values of independent variables which way that variable affects the final result. For instance, age on earnings, or age and education on earnings:

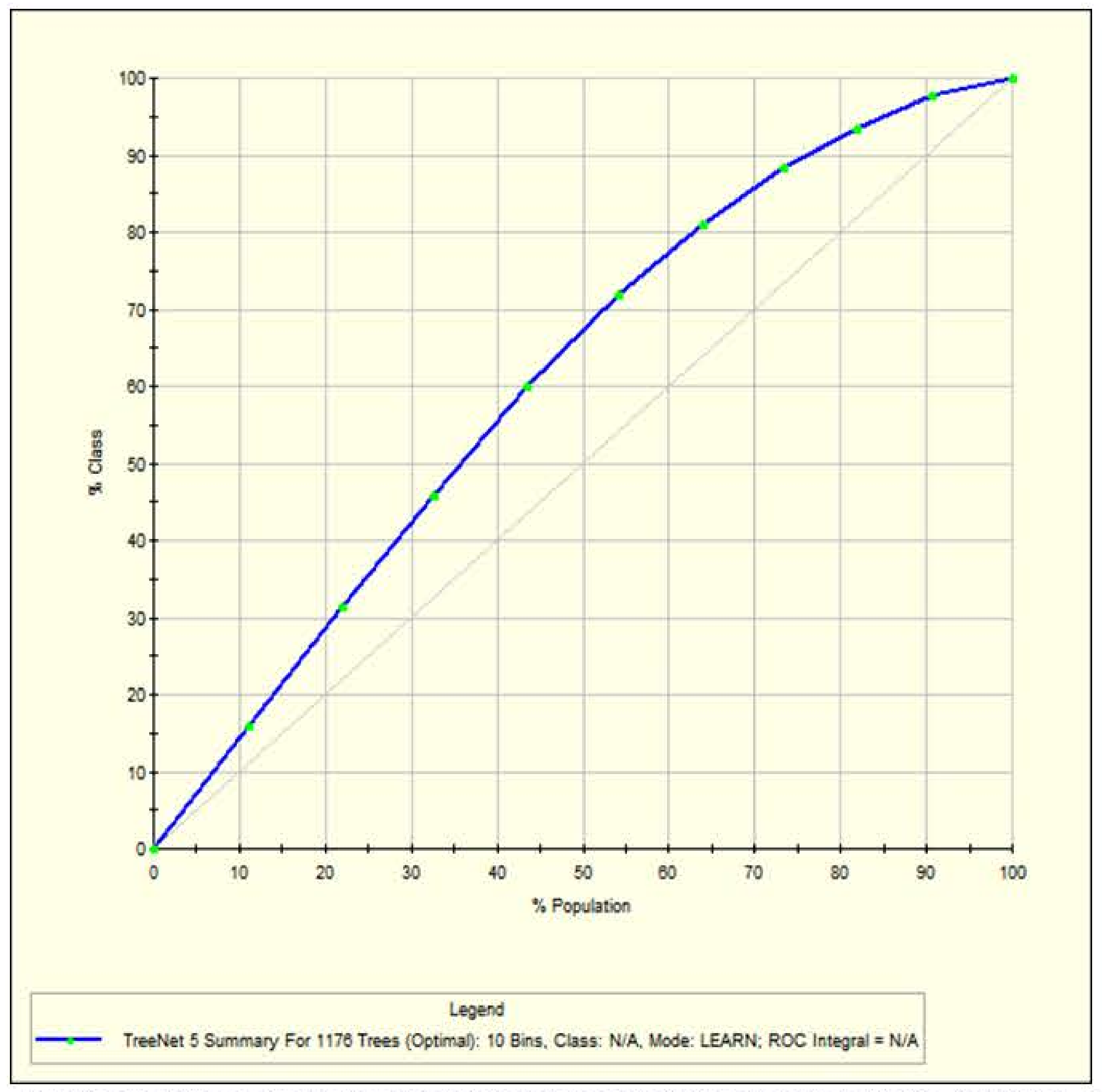




As we can see these graphs fit our expectations of the effects of age and education on earnings.

Another tool for evaluating results is the gains chart.

Gains Chart



On the x-axis we have the cumulative predicted dependent variable level for each observation highest to lowest (left to right) sorted by prediction as a percentage of the population. On the Y axis, at each of the points we have the summed percentage of their real dependent variable values as a percentage of the total dependent variable sum.

The first point shows that the top 11% of the predictions account for 16% of the real total of the dependent variable. More area between this line and the 45 º line, indicates the observations have been well sorted. If the line is exactly 45 º then the target is evenly distributed throughout the predictions and they may as well be random.

Our Project

Since TreeNet doesn't provide easily comprehensible models, or coefficients on its variables, the most pertinent use of TreeNet is its predictive accuracy. Professor Jacobsen and I have built three models to predict log hourly earnings. Two of these are OLS models, and one is a TreeNet model. Each one was built using two-thirds of the relevant data from the Current Population Survey, and tested on the remaining third of the data for predictive accuracy.

The variables that we used included veteran status, education, gender, potential years of experience, metropolitan status, white non-Hispanic, place of birth, region of the United States, number of people in the household, non-labor income, weeks of work missed, marital status, polynomial and interaction terms.

The simple model we built was constructed using variables that have been theoretically confirmed in the literature regarding earnings with no extraneous variables. The complex model was constructed to allow OLS to have the optimal model to compete with an algorithm that does that automatically. It was created by manually maximizing the adjusted R-squared.

We then compared our predicted results with the actual earnings and generated a Mean Squared Prediction Error for each of our models on both 2009 and 2010 data.

		2009			2010	
	TreeNet	Simple	Complex	TreeNet	Simple	Complex
Mean Squared Prediction Error	0.373	0.375	0.510	0.394	0.397	0.398
R-squared	0.334	0.321	0.359	0.314	0.303	0.320
Adjusted R-squared	-	0.315	0.349	-	0.302	0.313

TreeNet with default settings outperforms both OLS models. In fact we notice something similar to over-fitting in the complex linear model. We observe a higher R-squared value but a significantly worse MSPE on the 2009 data; this suggests that the R-squared is only relevant on the data that the regression fit to.

Professor Jacobsen and I have used this method to make predictions on men's and women's earnings, to estimate what women would make if they had the exact same characteristics as men. This method can therefore be used to generate adjusted earnings ratios similar to what is found in the existing earnings discrimination literature.

Model	2009	2010
Real Values Women over Men	0.77	0.79
TreeNet mean man treated as a women over average men's earnings	0.85	0.88

Thus TreeNet has produced similar results to those that we can produce using OLS, and the results may be preferable. This shows that using ensemble/neural net models can complement or even replace the work that has been traditionally done using standard regression models. In cases where there is less theoretical guidance regarding the form of the model, this methodology may be preferable because it automatically considers variable interactions and nonlinear forms.